5-2-2023

SMARTER CROWDWORK BY APPLYING SMART CONTRACTS?

Anna Hupe
University of Kassel, anna.hupe@uni-kassel.de

Ulrich Bretschneider
University of Kassel, bretschneider@uni-kassel.de

SMARTER CROWDWORK BY APPLYING SMART CONTRACTS?

Research in Progress

Anna Hupe, University of Kassel, Kassel, Germany, anna.hupe@uni-kassel.de

Ulrich Bretschneider, University of Kassel, Kassel, Germany, bretschneider@uni-kassel.de

Abstract

Crowdworking is characterized by flexibly engaging a workforce that is recruited for only one task. After the task is completed, the crowdworker and the company do not collaborate anymore. To identify the best-suited crowdwoker for a certain task, crowd working platform providers offer different tools to match both parties. Matching a crowdworker with the required skills for a task is an important factor to complete a crowdworking campaign successfully. As of today, the principal agent theorem and high transaction costs decrease the effectiveness of such campaigns due to information asymmetries between crodworkers and crowdsourcers. Based on a qualitative approach, we structurally develop a currently existing crowdworking campaign process. We then theoretically analyze how smart contracts can support and simplify the identified process while reducing the challenges of transaction cost economics and the principal agent theorem.

Keywords: Crowdworking Process, Smart Contracts, Information Asymmetry, Principal Agent Problem.

1 Introduction

The concept of outsourcing is indispensable in today's business world. Handing out tasks that lie outside a firm's core competences provides greater flexibility, better and/or cheaper results and allows to retain competitive. Next to traditional outsourcing, many organizations make use of crowdwork; a way of deploying labor flexibly and to benefit from swam intelligence (Durward et al., 2016a). Crowdwork is based on the basic pillars of crowdsourcing where tasks are outsourced to an anonymous crowd and reached via an open call on a crowdworking platform (Durward et al., 2016a). Throughout the whole process, crowdworkers stay anonymous as the open call, the allocation of tasks as well as the evaluation and payment are completed via IT-based platforms. Despite being able to reach a great diversity of candidates, the related inconspicuousness intensifies the principal agent problem. Due to the ad hoc work relationship that the three parties (crowdsourcers (handing out tasks), crowdworkers (overtaking tasks), platform providers (mediating)) enter, the conflict of interests and information asymmetry is pronounced. Conflict of interests lead to inefficiencies as the crowdsourcers, crowdworkers and crowdworking platforms want to maximize their individual benefit. Whereas the crowdsourcer (the "*principal*") wants to get tasks completed (at least) as good as doing it inhouse but less expensive, crowdworker (the "*agent*") targets to complete them with as little effort as possible and receive maximum remuneration.

This so-called principal-agent problem exists whenever one entity (the "*agent*") takes action on behalf of another entity (the "*principal*") (Arrow, 1986). As it is impossible to survive as an organization sustainably and competitively without outsourcing tasks, the principal-agent problem is omnipresent. Also, in an intraorganizational setting employees' (the agent) and management's (the principal) interests diverge where the principal depends on the agent's actions and does not have full information.

To reduce the principal agent theorem in employee-employer setting, management and psychological research has developed solutions such as performance-based payment (Grossman and Hart, 1992;

Haubrich, 1994). In the context of crowdworking, the classical employee-employer setting does not exist; it can rather be described as an ad hoc work relationship. Thereby, the divergence of interests and asymmetric information gets intensified; existent approaches to reduce the principal-agent problem do not apply. Making crowdworking campaigns more transparent for involved parties by bringing the process on a blockchain reduces asymmetric information. With crowdsourcers being able to access the crowdworker's work progress at any time and being sure about the originality of certificates that crowdworkers upload to the platform, information asymmetry between both parties is reduced to a minimum. Making use of the blockchain technology and smart contracts provides the relevant technology to make a crowdworking process more transparent, objective, and lean. With our research, we target to answer the following research questions: "*How does a crowdworking campaign process currently look like?*" and "*How would a crowdworking campaign process look like when including smart contracts?*".

First, we applied a qualitative approach to identify the conventional process of a crowdworking campaign. Subsequently, we examined the process to determine which steps can be modified through the implementation of smart contracts to reduce information asymmetry, moral hazard, and transaction costs. Smart contracts are computer programs or transaction protocols designed to automatically execute, control, or document legally relevant events and actions according to a contract or an agreement. These contracts operate on a blockchain, allowing for increased transaction transparency compared to conventional automation, by following predetermined if-then rules (Zou et al., 2021).

2 Theoretical Background

2.1 Information Asymmetry in Crowdwork

Crowdwork is a form of digital employment that is a subcategory of crowdsourcing. Crowdsourcing is composed of "*crowd*" and "*outsourcing*" and was introduced by Jeff Howe (2006). Whereas individuals in crowd*sourcing* campaigns are motivated intrinsically, crowd*workers* are only incentivized extrinsically; monetary inducements play a major role (Durward et al., 2016b). Crowdworkers have an interest in earning as much money in as little time as possible by completing tasks with as little effort (Al-Ani and Stumpp, 2016). This is because they do not perceive any value for themselves to complete tasks especially well; outstanding performance does not get remunerated additionally. The relationship between crowdsourcers and crowdworkers differs from a traditional worker-employee setting as crowdworkers do not receive an employment contract from crowdsourcers. There is only a verbal agreement between both parties. Legal obligations do not exist to the extent of a working contract as both only enter a contract with the crowdworking platform.

Crowdworking platforms take over the role of the intermediary and support crowdsourcers in defining tasks and solution requirements (Blohm et al., 2014). Platforms also handle the payment of crowdworkers and act as arbitration in case of conflicts. As with all intermediaries, also platforms need to be trusted and cannot guarantee full objectivity (Li, 2010). Despite those challenges that crowdworking bring along, it gets more and more popular among firms to make use of this concept. Handing out supportive tasks to the crowd allows full-time employees to focus more on value-creating activities. These are those tasks that a firm is generating most of its revenue and/or profit with (Groth and Kinney, 1994). The more tasks a crowdworker overtakes and the more crowdsourcers are satisfied with the results, the higher the chance that the crowdworker can request a higher financial compensation for completing tasks in the future (Ho et al., 2015). However, due to information asymmetry that exist between crowdworkers and crowdsourcers, the concept is characterized by inefficiencies. Including automation technologies and smart contracts provide possible solutions and increases efficiency in crowdwork campaigns (Eggers et al., 2021).

2.2 Smart Contracts

Automation technologies are widely applied by firms. Workflow management systems, enterprise resource management (ERM) systems, and robotic process automation are just a few examples of established automation technologies (Eggers et al., 2021). Even though Szabo defined *smart contracts* in 1996 for the first time, their current application fields are underrepresented in IS research (Eggers et al., 2021). According to Szabo "*A smart contract is a set of promises, specified in digital form, including protocols within which the parties perform on these promises*" (1996). In 2016, Tapscott and Tapscott extended Szabo's definition: "*smart contracts are computer programs that secure, enforce, and execute the settlement of recorded agreements between people and organizations*" (Tapscott and Tapscott, 2016). What Szabo (1996) and Tapscott and Tapscott (2016) agree on is that smart contracts are simple or more complex if-then-else relationships.

Whenever a situation and its associated action can be clearly described, the implementation of a smart contract is possible (Negara et al., 2021). The difference between incumbent automation technologies and smart contracts is that the secondary are programmed on a blockchain (Negara et al., 2021). The blockchain is a continuously updated, chronologically arranged, and publicly viewable register with information on ownership and transactions (Sahai and Pandey, 2020). Cryptographic principles ensure the retroactive immutability of entries and make smart contracts a very secure automation technology (Shailak Jani, 2020). In the context of smart contracts, the blockchain allows agreements to be visible for included contracting parties and being triggered automatically when predefined situations occur (Mohanta et al., 2018). This provides full transparency and makes intermediaries redundant (Lin et al., 2020). This increases efficiency, security and creates trust between involved contracting parties.

Putting a smart contract into practice can be divided into four steps: creating, deploying, executing, and completing (Shailak Jani, 2020) (see figure 1).

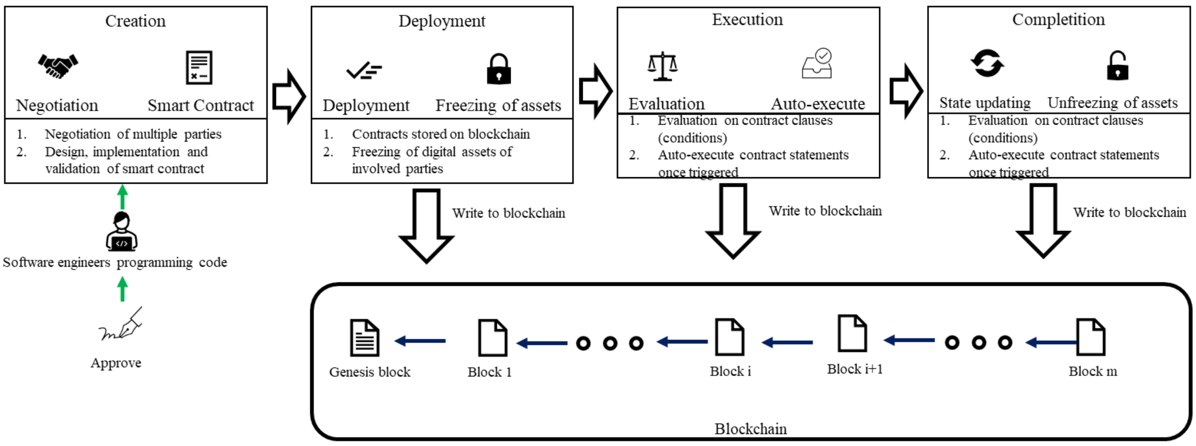

Figure 1. Setting up a Smart Contract, inspired by Shailak Jani (2020).

As a first step, the parties involved in the contract agree on contingencies and contractual terms. It is relevant to include all possible occurrences and define executions that should be triggered by the contract (Hu et al., 2020). After included parties come to an agreement and completed their negotiation, the smart contract gets programmed accordingly on a blockchain. The most common blockchain for smart contracts is Ethereum (Eggers et al., 2021). Having smart contracts written on Ethereum provides retrospective immutability of transactions (Kim and Shin, 2019). An exact (pre) specified execution is guaranteed which provides security to involved parties. Whenever the agreed terms and/or situation(s) occur, the smart contract completes the related and predefined action(s) (Shailak Jani, 2020).

Smart contracts are not only transparent and reduce the principal-agent problem but also automate business processes visible to all contract partners (Eggers et al., 2021). In a crowdworking campaign setting, crowdworkers and crowdsources would then have the same information (no information asymmetry). This makes intermediaries redundant. Intentions for bypassing middlemen are diverse and often relate to financial, time, and/or political reasons (Shahab and Allam, 2020). Financial, time and efficiency reasons are decisive to organize crowdworking campaigns via smart contracts. Third parties that are currently mediating between contracting parties and that need to be trusted by included contract partners can be replaced by smart contracts. Those immutable computer protocols create the trust needed for ensuring a sustainable business relationship.

After a smart contract is programmed (step 1), it gets deployed. This second step includes storing the smart contract on a blockchain which makes it transparent for all included parties (see figure 1). Every transaction and completion of the smart contract is (as typical for the blockchain) listed in the genesis block and thus can be retraced at any time (Shailak Jani, 2020). This technology provides the previously mentioned transparency and security. In the execution phase (step 3), the smart contract consistently checks whether predefined situations appear (e.g., crowdworker completed a task satisfactory). As soon as the contract identifies a defined condition, it executes the connected task (e.g., transferring remuneration to crowdworker's account) (Christidis and Devetsikiotis, 2016). The final step, the completion of the transaction (step 4), allocates the digital assets that are triggered by the smart contract. Our research attempts to identify in which steps of a crowdworking campaign smart contracts can be implemented to improve the process and to overcome existing challenges such as trust issues between related parties, information asymmetry and conflict of interest.

3 Research Design

As our study has two research objectives (1) identifying a crowdwork campaign process and (2) identifying applications for smart contracts in the process, our research design is two-fold. To determine a process model of a crowdwork campaign, we conducted a qualitative study. Carrying out six interviews with managers of six different crowdworking platform providers allowed us to get extensive insights into their crowdworking platforms' processes, challenges, and difficulties in their everyday business. Half of our interview partners are working for a crowdworking platform offering microtasks, the other three interview partners work for platforms that specialize in macrotasks. All interviews were conducted online, took around one hour, and were recorded with prior consent.

Following Carroll (2000), we deploy a five-stage analysis process after having transcribed the interviews. In the first stage, we break down the interview transcripts that we gained from the collected data. We then subsequently assign each unit with specific codes by means of MAXQDA. In the second stage, we analyze the coded units. We investigate categories of related issues, and sort and cluster them accordingly (code aggregation). In the third stage, we test whether these categories are inter-subjectively resistant (Carroll 2000) by creating a coding scheme where categories and exemplary indicators are defined. In the fourth stage, we intensively discuss the results from the previous step to either (1) build a consensus, (2) drop them from further analysis, or (3) create a further category. Having authors with different backgrounds and experiences discussing the categories ensures greater objectivity of the results despite the approach of qualitative research. During the axial coding process (fifth stage), we aim at finding plausible relationships between all identified categories, thereby organizing the theoretical components into higher-level (or core) categories (Olsson et al., 2008).

Our data collection process is not completed yet, we are planning on having 14 more interviews with crowdworking platform providers, crowdworkers and crowdsourcers to get a broad and extensive picture of the current process. This also ensures the validity and credibility of our study. With our research in progress, we identify a crowdworking campaign process as well as challenges that arise throughout the process. For overcoming those challenges, we take the identified process as a basis and theoretically develop how smart contracts can help to overcome those problems. We thereby consult literature on smart contracts, blockchains, and process management. For future research are planning to include a DSR approach in order to validate our findings. In that context we will also precisely include Carvalho and Karimi (2021) decision model and Pedersen et al. (2019) decision path.

4 Preliminary Findings

4.1 Process Model of a Crowdwork Campaign

After analyzing the data, we develop the following five-phase crowd working campaign process (figure 2). Before the process starts, crowdworkers need to create an account on the relative platform (interviewee 4,5). Creating an account includes indicating their qualifications, degrees, special knowledge, and skills. Certificates proving that information are also uploaded to the platform. To validate the stated qualifications, crowdworking platforms ask crowdworkers to complete tests before their profile goes live. To ensure a realistic reflection of their performance, crowdworkers are not aware that the task conducted is a test ("*We ensure that crowdworkers do not know that they are completing a trial task*", interviewee 3). Despite these quality assurance mechanisms, the originality of certificates is not assured; they can easily be fake without crowdsourcers and/or platforms realizing (interviewee 1, 5, 6). Checking and ensuring the authenticity of uploaded certificates presents a great challenge for platform providers as those evaluations given by previous contracting authorities are often subjective (interviewees 1, 2, 3, 4).

Figure 2 depicts the five-phase model of a crowdwork campaign that we developed based on the interviews. In the first phase, the *initiation phase*, the crowdsourcer defines a task, precisely describes it, decides on a crowdworking platform, and prepares an open call ("*Our goal is to only have job descriptions uploaded that are clearly defined to ensure the best fit with the crowdworker*", interviewee 6). In this phase, it is extremely relevant for crowdsourcers to decide on the correct platform as different crowdworking platforms impart different crowdworkers (interviewees 1, 4). In the second phase (*bidding phase*) and after publishing an open call, platforms either (1) ask crowdworkers to create proposals for completing the uploaded task (macro tasks) (interviewees 1, 3, 5), (2) crowdworkers pick the task to be completed (microtasks) (interviewees 2, 4, 6), or (3) crowdworkers apply for executing the task (macrotasks) (interviewees 1, 3, 5). The third phase only exists for macro tasks and is about deciding on a crowdworker. This *decision phase* either starts with (1) crowdworkers submitting their proposals followed by the crowdsourcer comparing and rating all results or with (2) crowdsourcers

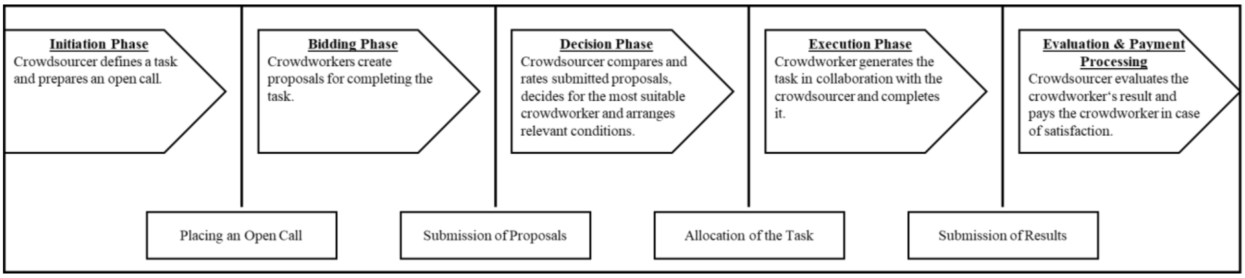

Figure 2. Five-phase Process Model of a Crowdwork Campaign.

deciding which crowdworker is most suitable for a task (interviewees 1, 3). Based on the competence profile of the crowdworker, and the rating, the platform offers recommendations for crowdsourcers. These recommendations/matching principles allow platforms to present recommendations of which crowdworker is suitable for a certain task ("*This matching presents great value for crowdsourcers as they get offered the best candidates that fit a certain task*", interviewee 1). The decision basis for the most suitable crowdworker is often a combination of demographic characteristics (e.g., age, gender, place of residence), skills (e.g., IT expertise, language skills), and previously completed tasks (interviewees 1, 3). In the penultimate phase, the *execution phase*, tasks are allocated and completed by the chosen crowdworker (macrotasks) or the crowdworker that overtook the microtask. If necessary, additional information is shared with the crowdworker to fulfill the task (interviewee 3). After the results are completed, they are submitted via the platform. Crowdsourcers then can ask for changes and adaptions ("*In that step, crowdsourcer have the chance to ask for improvements and edits. The only requirement is that they are precisely stated in the initial task description*", interviewee 5). Having submitted the solutions satisfactorily induces the *evaluation and payment processing phase*. In this final phase, the crowdsourcer evaluates the crowdworker's performance, communication etc., during the handling time. Crowdsourcers rank them into a one-to-five-star ranking (interviewees 1, 2, 3, 4, 5, 6). Notwithstanding the above, crowdworkers receive their payment ("*Crowdsourcers can only upload tasks if they have enough liquidity uploaded to the platform. We then subtract the value of the task and transfer it to the crowdworker's account as soon as a task is completed satisfactorily or after the expiration of the reviews. This ensures that crowdworkers get paid*", interviewee 1).

The process of a crowdworking campaign can be described as repetitive. However, many crowdworking platform providers do not make use of automatization as but have employees that manually match crowdworkers and crowdsourcers (interviewees 3, 5). This is because so far tested algorithms do not provide the initially defined support for crowdworking platforms.

To maintain crowdworking as an appealing and cost-effective concept for companies, the integration of automation technologies is crucial. However, current automation solutions have been inadequate, prompting the exploration of smart contract solutions and their potential value in the crowdworking process (interviewee 3, 5). We theoretically identify areas in the crowdworking process where smart contracts can be implemented to improve the mechanism. Advantages of smart contracts over traditional automation include expediting the matching of crowdworkers and crowdsourcers, decreasing principal-agent and moral hazard issues due to complete transparency, and reducing administrative costs for intermediaries. Payment processes are also automated and secured by the blockchain, which not only minimizes manual effort but also mitigates the principal-agent theorem and moral hazard concerns between parties. By utilizing smart contracts, the attractiveness of the crowdworking concept is improved for both crowdworkers and crowdsourcers.

4.2 Application of Smart Contracts in Crowdwork Campaign Processes

To ensure long term success and competitiveness in a fast-changing business environment, organizations focus on those tasks that create most value. Having internal employees focusing on tasks that provide a unique selling proposition which then leads to a competitive advantage (Barney, 1995), implies that other tasks need to either be automated or outsourced. It should be noted that whenever a task is handed out and an action is taken on behalf of a different entity, a conflict-of-interest results (Boatright, 1992).

This popular principal agent theorem is omnipresent in any interpersonal/interorganizational relation as the executor (the agent) has different interests than the organization (the principal). The main interest of a principal handing out a task to an agent is to get tasks solved (not less than) as good as if they were completed inhouse, but more economical. In contrast, the agent has the interest to complete a task with as little effort as possible while receiving maximum remuneration (Boatright, 1992). Thus, the interests of the principal and the agent diverge. Next to the conflict of interest, a principal-agent relationship is characterized by asymmetric information (Stiglitz, 1974). Despite clear agreements and contracts, the principal cannot observe the agent's actions in full detail. Even if the principal would control the agent during the working process, he will never be aware of whether the agent could have completed a task faster or better (Stiglitz, 1974).

Management research offers various mechanisms to align the interests of agents with those of principals and reduce asymmetric information (Grossman and Hart, 1992; Haubrich, 1994). These include financial incentives like success-based commissions, profit sharing, and performance measurement, as well as non-monetary inducements like group commitment measures (Chan and Lam, 2011; Miller, 2005). As our research focuses on crowdwork, traditional mechanisms to align the interests of agents and principals do not apply due to anonymity and ad hoc work relationships.

Anonymity aggravates the principal-agent problem, as crowdworkers are only interested in maximizing their self-interest and completing tasks quickly (Amanor-Boadu and Starbird, 2005). Platforms act as intermediaries and ensure quality control by only paying crowdworkers if crowdsourcers are satisfied with the delivered solutions (see evaluation & payment processing phase in figure 2). However, according to interviews, crowdsourcers often spend little time screening solutions, which can lead to crowdworkers being paid despite incomplete or unsatisfactory tasks (interviewee 3, 5).

In the following, we demonstrate how including smart contracts in the above developed process reduces the principal agent problem between crowdworkers, crowdsourcers and crowdworking platform providers. We identify additional advantages that smart contracts offer when including them in the process of a crowdwork campaign. Our research in progress depicts that including smart contracts in the crowdworking process provides greater transparency, objectivity, and efficiency. This makes it possible to build trusting relationships between crowdworker and crowdsourcer whose relationship in a traditional setting is characterized by distrust. In contrast to smart contracts, traditional automation technologies do not provide solutions to the issue of distrust.

With smart contracts screening task descriptions uploaded by crowdsourcers, identifying relevant keywords, and proposing the best candidates based on their profiles makes the whole process leaner. The decision phase (phase 1 & 2 of the five-phase process) gets more transparent by applying smart contracts (see figure 3). As the origin of certificates as well as their originator can be retraced due to the transparency of the underlying blockchain technology, fakes can be immediately detected (Golosova and Romanovs, 2018; Yasin and Liu, 2016). Giving permission to the crowdsourcer to examine the crowdworker's documents can be one requirement to apply for a task (Christidis and Devetsikiotis, 2016). The information asymmetry concerning skills existent in a traditional crowdworker crowdsourcer setting can be diminished with this measure. Based on certificates, experience, and further relevant information that the smart contract has access to, the algorithm objectively proposes one or a predefined number of crowdworkers. Depending on what crowdsourcers wish for and how the smart contract is programmed, contracting authorities either can select among the proposed ones or leave the decision with the algorithm. Having access to all relevant information and being clearly trained on what to choose in which situations (if-then-relationships), smart contracts operate completely objectively (Kim and Laskowski, 2017). Additionally, the automatization is expeditious; suitable crowdworkers can be identified within seconds. At the same time smart contracts are more secure than traditional automation solutions due to its storage on a blockchain. As crowdworkers play the major role in the execution phase (phase 4), smart contracts find application in automatically handing over additional information (e.g., via an oracle) required for completing a task. This process is triggered as soon as a crowdworker is determined. Letting smart contracts share additional information, reduces the manual effort for crowdsourcers. Depending on the complexity of the task given to crowdworkers, the handed in solution can either be reviewed by a crowdsourcer's employee (complex task) or by the algorithm itself (simple task). Combining the review phase with first having a dispassionate smart contract controlling a task and then letting an internal employee screening the solution is a further option that reduces the crowdsourcer's effort. Dual quality control increases the chance that tasks given to the crowd comply with the crowdsourcer's quality expectations.

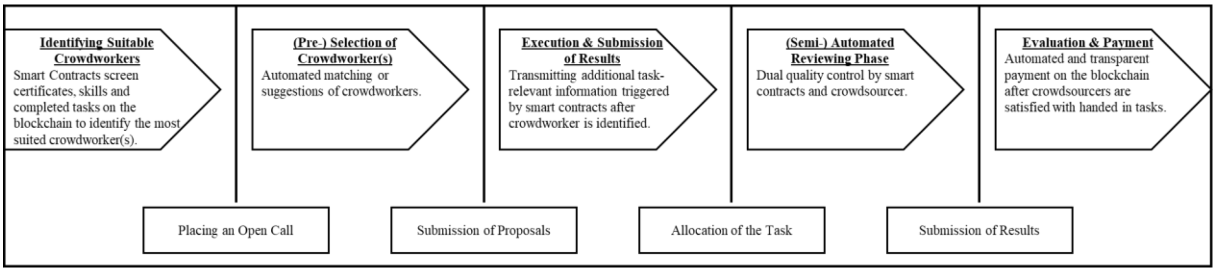

Figure 3. Process Model of a Crowdwork Campaign When Including Smart Contracts.

Making the whole crowdworking process completely transparent by relocating it to a blockchain and applying smart contracts eliminates the principal agent problems main challenges (see figure 3). This reduces information asymmetry and moral hazard due to traceability of all actions on the blockchain.

Matching mechanisms are exclusively objective and transparent. At the same time, smart contracts offer the feature to prioritize different projects for the crowdworker according to predefined criteria.

As neither crowdworkers nor crowdsourcers need to trust a middleman as all terms and conditions are clearly defined in the smart contract, those computer protocols have the potential to transform and improve the existent process. Also, the payment procedure is simplified. For this, a smart contract gets programmed in a way that the payment gets triggered autonomously as soon as crowdsourcers accept the solutions handed in by crowdworkers. Having all transactions indicated on the blockchain, eliminates uncertainties between crowdworkers and crowdsourcers. Evaluations as well as completed payments are also visible for included parties (Lindman et al., 2017). Including smart contracts in the crowdworking campaign provides transparency which then leads to security as information asymmetry is reduced (Zaefarian et al., 2017).

5 Conclusion and Next Steps

Our preliminary findings demonstrate that smart contracts have the potential to transform the crowdworking process by providing a decentralized, immutable, and transparent way to automate tasks and transactions. This can address key challenges such as information asymmetry, where the crowdworker may not have enough information about the task or employer, and moral hazard, where the crowdworker may not be incentivized to perform their work to the best of their ability.

We developed a mechanism based on expert interviews to improve the current process of crowdworking campaigns. This involved identifying the current process, highlighting its drawbacks and challenges, and proposing an improved process with the implementation of smart contracts. Our study is unique in that it is the first to present specific application possibilities for smart contracts in crowdworking campaigns. This contribution to literature on online labor platforms and smart contracts can help advance the field by providing new insights and avenues for research.

Our data analysis suggests that implementing smart contracts in crowdworking campaigns can reduce costs and increase transparency, making crowdwork more attractive to all parties involved. Key challenges in the crowdworking setting such as the principal-agent theorem, information asymmetry and conflict of interest are reduced due to transparent blockchain solutions. For example, the use of smart contracts can make it easier for crowdsourcers to match crowdworkers to specific tasks, while providing a transparent record of the selection and working process. This can lead to a larger base of crowdworkers, which increases the chances of crowdsourcers finding an appropriate worker for their task.

With our research we contribute to current crowdwork and online labor platform literature, as well as to application opportunities for smart contracts. IS research has mainly focused on smart contract use cases in the finance and insurance industry so far (Eggers et al., 2021). Our research combines online labor platforms with the opportunities that blockchains and smart contracts provide. By conducting expert interviews, we present the currently applied process and highlight its drawbacks and challenges.

To further strengthen our study, we plan additional interviews with crowdworking platform providers, as well as with crowdworkers and crowdsourcers to identify what is important to them when smart contracts are implemented. Additionally, we plan to follow a design science research approach to increase the rigor of our study.

As the application of smart contract in online labor platform setting is novel, further research is needed to determine potential challenges and solutions to implementing smart contracts in crowdworking campaigns. This includes addressing privacy and legal challenges, technical barriers, and analyzing the acceptance of smart contracts and reasons for reluctance. By addressing these challenges, the field of crowdworking can continue to evolve and benefit from the advantages of smart contracts. Our study provides a first important model to indicate the potentials and relevance for smart contracts in crowdworking campaigns.

6 References

Al-Ani A and Stumpp S (2016) Rebalancing interests and power structures on crowdworking platforms. *Internet Policy Review* 5(2).

Amanor-Boadu V and Starbird S (2005) In search of anonymity in supply chains. *Journal on Chain and Network Science* 5(1): 5–16.

Arrow KJ (1986) Chapter 23 Agency and the market. In: Elsevier, pp. 1183–1195.

Barney JB (1995) Looking inside for competitive advantage. *Academy of Management Perspectives* 9(4): 49–61.

Blohm I, Leimeister JM and Zogaj S (2014) Crowdsourcing und Crowd Work – ein Zukunftsmodell der IT-gestützten Arbeitsorganisation? In: Brenner W and Hess T (eds) *Wirtschaftsinformatik in Wissenschaft und Praxis:* Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 51–64.

Boatright JR (1992) Conflict of Interest. In: Oxford University Press, pp. 187–203.

Carvalho A and Karimi M (2021) Aligning the interests of newsvendors and forecasters through blockchain-based smart contracts and proper scoring rules. *Decision Support Systems* 151: 113626.

Chan KW and Lam W (2011) The trade-off of servicing empowerment on employees' service performance: examining the underlying motivation and workload mechanisms. *Journal of the Academy of Marketing Science* 39(4): 609–628.

Christidis K and Devetsikiotis M (2016) Blockchains and Smart Contracts for the Internet of Things. *IEEE Access* 4: 2292–2303.

Durward D, Blohm I and Leimeister JM (2016a) Crowd Work. *Business & Information Systems Engineering* 58(4): 281–286.

Durward D, Blohm I and Leimeister JM (2016b) Rags to Riches - How Signaling Behaviour Causes a Power Shift in Crowdsourcing Markets. *SSRN Electronic Journal*. DOI: 10.2139/ssrn.3159165.

Eggers J, Hein A, Weking J, et al. (2021) Process Automation on the Blockchain: An Exploratory Case Study on Smart Contracts. In: *Proceedings of the 54th Hawaii International Conference on System Sciences: (*ed T Bui). Hawaii International Conference on System Sciences.

Golosova J and Romanovs A (2018) The Advantages and Disadvantages of the Blockchain Technology. In: *2018 IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE):* Vilnius, 8 - 10 November 2018, pp. 1–6. IEEE.

Grossman SJ and Hart OD (1992) An Analysis of the Principal-Agent Problem. In: Cummins JD, Dionne G and Harrington SE (eds) *Foundations of Insurance Economics:* Dordrecht: Springer Netherlands, pp. 302–340.

Groth JC and Kinney MR (1994) Cost Management and Value Creation. *Management Decision* 32(4): 52–57.

Haubrich JG (1994) Risk Aversion, Performance Pay, and the Principal-Agent Problem. *Journal of Political Economy* 102(2): 258–276.

Ho C-J, Slivkins A, Suri S, et al. (2015) Incentivizing High Quality Crowdwork. In: *Proceedings of the 24th International Conference on World Wide Web: (*eds A Gangemi, S Leonardi and A Panconesi), Florence Italy, 18 05 2015 22 05 2015, pp. 419–429. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee.

Hu K, Zhu J, Ding Y, et al. (2020) Smart Contract Engineering. *Electronics* 9(12): 2042.

Kim H and Laskowski M (2017) A Perspective on Blockchain Smart Contracts: Reducing Uncertainty and Complexity in Value Exchange. In: *2017 26th International Conference on Computer*

Communication and Networks (ICCCN): Vancouver, BC, Canada, 31 July - 3 August 2017, pp. 1–6. IEEE.

Kim J-S and Shin N (2019) The Impact of Blockchain Technology Application on Supply Chain Partnership and Performance. *Sustainability* 11(21): 6181.

Li W (2010) Peddling Influence through Intermediaries. *American Economic Review* 100(3): 1136–1162.

Lin H, Yang Z, Hong Z, et al. (2020) Smart Contract-based Hierarchical Auction Mechanism for Edge Computing in Blockchain-empowered IoT. In: *2020 IEEE 21st International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM):* Cork, Ireland, 31 August - 3 September 2020, pp. 147–156. IEEE.

Lindman J, Tuunainen VK and Rossi M (2017) Opportunities and Risks of Blockchain Technologies: A Research Agenda. In: *Proceedings of the 50th Hawaii International Conference on System Sciences (2017):* Hawaii International Conference on System Sciences.

Miller GJ (2005) Solutions to Principal-Agent Problems in Firms. In: Menard C and Shirley MM (eds) *Handbook of New Institutional Economics:* Berlin/Heidelberg: Springer-Verlag, pp. 349–370.

Mohanta BK, Panda SS and Jena D (2018) An Overview of Smart Contract and Use Cases in Blockchain Technology. In: *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT):* Bangalore, 10 - 12 July 2018, pp. 1–4. IEEE.

Negara E, Hidayanto A, Andryani R, et al. (2021) Survey of Smart Contract Framework and Its Application. *Information* 12(7): 257.

Olsson, Conchúir, Ågerfalk, et al. (2008) Two-Stage Offshoring: An Investigation of the Irish Bridge. *MIS Quarterly* 32(2): 257.

Pedersen AB, Risius M and Beck R (2019) A Ten-Step Decision Path to Determine When to Use Blockchain Technologies. *MIS quarterly executive*: 99–115.

Sahai A and Pandey R (2020) Smart Contract Definition for Land Registry in Blockchain. In: *2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT):* Gwalior, India, 10 - 12 April 2020, pp. 230–235. IEEE.

Shahab S and Allam Z (2020) Reducing transaction costs of tradable permit schemes using Blockchain smart contracts. *Growth and Change* 51(1): 302–308.

Shailak Jani (2020) Smart Contracts: Building Blocks for Digital Transformation, Unpublished.

Stiglitz JE (1974) Incentives and Risk Sharing in Sharecropping. *The Review of Economic Studies* 41(2): 219.

Szabo N (1996) Smart Contracts: Building Blocks for Digital Markets. *EXTROPY: The Journal of Transhumanist Thought* 18(2).

Tapscott D and Tapscott A (2016) *Blockchain revolution: How the technology behind Bitcoin is changing money, business and the world*. New York, New York: Portfolio/Penguin.

Yasin A and Liu L (2016) An Online Identity and Smart Contract Management System. In: *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC):* Atlanta, GA, USA, 10 - 14 June 2016, pp. 192–198. IEEE.

Zaefarian G, Thiesbrummel C, Henneberg SC, et al. (2017) Different recipes for success in business relationships. *Industrial Marketing Management* 63: 69–81.

Zou W, Lo D, Kochhar PS, et al. (2021) Smart Contract Development: Challenges and Opportunities. *IEEE Transactions on Software Engineering* 47(10): 2084–2106.